

Prova scritta 29 Gennaio 2015

Esercizio 1

```
void insordinato(elem*& p0, int k) {
    elem *r, *s;
    for (s=p0; s!=NULL && s->info<k; s=s->pun)
        r= s;

    if (s!=NULL && s->info == k)
        return;

    elem = t = new elem;
    t->info = k;

    if (s==p0) {
        t->pun = p0;
        p0 = t;
    }
    else {
        r->pun = t;
        t->pun = s;
    }
}

void modificaLista(elem*&L, elem*P) {
    for (elem *q =P; q!= NULL; q=q->pun)
        insordinato(L, q->info);
}
```

Esercizio 3

```
bool ordinatoRic(int v[], int n) {
    if (n==1)
        return true;
    if (v[n-1] < v[n-2])
        return false;
    return ordinatoRic(v, n-1);
}
```

Esercizio 2

```
bool controlla (char* m, int r, int c,
                const char* nome) {

    ifstream in (nome);
    if (!in) return false;
    int n;
    in>>n;
    if (!in) return false;
    in.close();
}
```

```

bool trovato;
char a;

if (n <= c){
    for (int i=0; i<r; i++)
        for(int j = 0; j<= c-n; j++) {
            a = m[i*c+j];
            trovato = true;
            for (int conta=1; conta<n; conta ++ )
                if(m[i*c + j + conta]!=a)
                    trovato = false;

            if (trovato)
                return true;
        }
}

if (n <= r){
    for (int j=0; j<c; j++)
        for(int i = 0; i<= r-n; i++) {
            a = m[i*c+j];
            trovato = true;
            for (int conta=1; conta<n; conta ++ )
                if(m[i*c + j + conta]!=a)
                    trovato = false;

            if (trovato)
                return true;
        }
}

return false;
}

```

Esercizio 4

4.1

Raggruppando le cifre tre a tre e decodificando il singolo gruppo si ottiene:

111-110
 7 6

Ossia 76 in base 8

L'intero associato è il -2,
 essendo la rappresentazione
 111-110 quella precedente alla 111-111,
 che è noto essere la rappresentazione di -1.

Un altro modo per scoprire che era la rappresentazione di -2 era quello di utilizzare la formula: $2^p - \text{abs}(A)$:
 $2^6 = 64 - (56 + 6) = 64 - 62 = 2$ (dunque -2, perché sapevamo essere un numero negativo in quanto iniziava per 1).

4.2

Sia X la penultima cifra e Y l'ultima cifra del proprio numero di matricola.

Consideriamo per il momento il caso particolare $X = 4$ e $Y = 5$.

Il programma inizializza AL utilizzando X e Y come fossero cifre esadecimali, dunque il AL ci finira' 0100-0101 (0x45)

L'AND con 0x0F produce 0x05 (azzerata la cifra X, qualunque essa sia).

L'OR con 0x30 trasforma in un 3 la prima cifra esadecimale, lasciando inalterata la seconda.

(Dunque, indipendentemente dal valore di X, arrivati a questo punto AL conterra' sempre 0x3Y.)

La call output stampera' a video il carattere '5', essendo '5' proprio il carattere ASCII associato all'esadecimale 0x35. (In generale stampera' a video la cifra meno significativa 'Y').

Il successivo shift a destra setta il carry flag solo nel caso in cui la cifra Y sia dispari, nel qual caso il programma stampa anche la lettera "D".

In definitiva il programma stampera' a video "YD" nel caso di Y dispari e "Y" altrimenti, (lo zero viene considerato pari):

Y=0 stampa "0"
Y=1 stampa "1D"
Y=2 stampa "2"
Y=3 stampa "3D"

ecc...